We have developed a NoSQL project using C# 6.0 language and .Net Framework in visual studio 2015.

In first project-OCD we have discussed about the package structure of the project and activity diagram and data flow for this project. After implementing the project we can see that we have almost completely developed a similar model with just slight changes in package structure or data flow between packages.

We have textExec which is our main package and all other packages are directly or indirectly called from this package. DBengine stores a dictionary which is our <key,value> database. Value is an element given by DBElement<metadata,payload>, where metadata gives information about the payload i.e. actual value of that specific database entry. It includes name, description, timestamp and dependency of that element with other elements.

QueryEngine package is used to do queries on DBEngine using query predicates. For these query predicates we have used lambda functions. As discussed in OCD for each query result we are creating an immutable database DBFactory which is generated using only List of keys from result of query on Database and a reference to database. For, compound queries we can use list of query predicates each containing a lambda function which defines a simple query that is to be applied on the database. Thus, by applying simple queries on DBFactory one after another, we can make it work as compound query.

Using PersistEngine package we are writing data from database to XML file using XDocument class from System.linq package.

In the original concept which I developed during project 1 – OCD, I have used XMLParser package to parse xml files and add data to our database. But, after implementation of project we can see that, we are able to do both writing data from database to xml file and from xml file to database both can be done in same package.

We have developed a Scheduler class using Scheduler form System.Timers package to continuously persist data to XML files after specific time interval.

I have added EventLogger package in my OCD to log each operation that is executed in any of the packages. This package I have not added into the project implementation.

Also, I used to Errorhandler package to handle any types of error in any package, but after implementation I can see that it can be done directly in each package and we don’t need an extra package for that.

Concepts which were thought for this project and included in OCD, such that sharding and compound queries were not implemented in this project.